ON STATE-SYNCHRONIZED AUTOMATA SYSTEMS

Jiří Kučera, <ikucera@fit.vutbr.cz>

October 10, 2013

ABSTRACT

A state-synchronized automata system (SCAS) of degree n, first introduced and investigated in [1, 2], consists of n components, which are in general pushdown automata, and a finite control language. For some SCAS of degree n, Γ, every word from control language has the form $q_1q_2 \ldots q_n$, where q_i denote some state of the i-th component of Γ, $1 \leq i \leq n$. The computation step in Γ is performed if and only if every i-th component of Γ can perform the move from its current state q_i to its new state q'_i, $1 \leq i \leq n$, and there are two words $q_1q_2 \ldots q_n$ and $q'_1q'_2 \ldots q'_n$ in the control language of Γ. The language accepted by some SCAS is the language accepted by its first component, which is the only component that can read the input tape.

From the hierarchy of formal languages point of view, there are no new language families described by SCAS. It was proven in [2] that the family of languages described by SCAS with at least two pushdown automata is equivalent to the family of recursively enumerable languages. Similarly, the family of languages described by SCAS with exactly one pushdown automaton and arbitrary number of finite automata coincide with the family of context free languages.

The further investigation of SCAS will be focused on determinism in these systems, mainly on proving Conjecture 7.1 in [2] which says that the family of languages described by deterministic SCAS (dSCAS) with at least two pushdown automata and the family of recursively enumerable languages coincide. To find out whether the given SCAS of degree n is deterministic or not, it must be firstly converted to n-pushdown automaton (the general form of two-pushdown automaton, see [3]).

After the investigation of determinism, the other ways of component synchronization will be considered, with the respect to control language. This step is implied by the need to decrease the SCAS power.

The SCAS can be used as the formal model for parallel compilers, but there are many other possible areas of application of these systems.

REFERENCES

[1] KUČERA, Jiří. On state-synchronized automata systems. **In:** DRAHANSKÝ, M. and ORSÁG, F., eds. *Proceedings of the 19th Conference STUDENT EEICT.* 2013, vol. 2, p. 216–218. ISBN 978-80-214-4694-6.

[2] KUČERA, Jiří. *A combination of automata and grammars.* Brno (Czech Republic), 2013. Master's thesis. Brno University of Technology, Faculty of Information Technology, Department of Information Systems.

[3] MEDUNA, Alexander. *Automata and Languages: Theory and Applications.* Springer, 2000. ISBN 81-8128-333-3.